Name: Ryan Hartman Date: 9/17/19

Point values are assigned for each question. Points earned: ____ / 100, = ____ %

1. Find an upper bound for $f(n) = n^4 + 10n^2 + 5$. Write your answer here: **$2n^4$, or $O(n^4)$)** (4 points)

   Prove your answer by giving values for the constants c and n_0. Choose the smallest integral value possible for c. (4 points)

$$n^4 + 10n^2 + 2 \leq cn^2 \quad \forall\, n \geq n_o$$

$$n^4 + 10n^2 + 2 \leq 2n^4 \quad \forall\, n \geq 4$$

$$\boxed{\therefore\ \boldsymbol{C = 2, n_o = 4}}$$

2. Find an asymptotically tight bound for $f(n) = 3n^3 - 2n$. Write your answer here: **$\theta(n^3)$** (4 points)

   Prove your answer by giving values for the constants c and n_0. Choose the smallest integral value possible for c. (4 points)

   Check Lower bound:

$$3n^3 - 2n \geq cn^3 \quad \forall\, n \geq n_o$$

$$3n^3 - 2n \geq 2n^3 \quad \forall\, n \geq 2$$

$$\boxed{\therefore\ \boldsymbol{C_1 = 2}}$$

   Check upper bound:

$$3n^3 - 2n \leq cn^3 \quad \forall\, n \geq n_o$$

$$3n^3 - 2n \leq 3n^3 \quad \forall\, n \geq 1$$

$$\boxed{\therefore\ \boldsymbol{C_2 = 3 \qquad C_1 = 2 \qquad n_o = 2}}$$

3. Is $3n - 4 \in \Omega(n^2)$? **<u>No.</u>**

   If yes, prove your answer by giving values for the constants c and n_0. Choose the smallest integral value possible for c. If no, derive a contradiction. (4 points)

For $3n - 4 \epsilon \Omega(n^2)$, there must exist positive constants n_o and C such that:

$$3n - 4 \geq Cn^2 \qquad \forall n \geq n_o$$

Well, if we simply solve this expression for n:

$$-4 \geq Cn^2 - 3n$$

$$n(n-3) \leq -\frac{4}{C}$$

It becomes clear that the constants n_o and C *cannot* exist. A quantity of n cannot be less than or equal to some constant, because n must be able to grow indefinitely. This inequality implies that n must be restricted to be below a constant, which means no n_o can be chosen, and proves by contradiction that this is **not** of the order $\Omega(n^2)$.

4. Write the following asymptotic efficiency classes in **increasing** order of magnitude.
   $O(n^2)$, $O(2^n)$, $O(1)$, $O(n \lg n)$, $O(n)$, $O(n!)$, $O(n^3)$, $O(\lg n)$, $O(n^n)$, $O(n^2 \lg n)$ (2 points each)

$O(1),\ O(lg\ n),\ O(n),\ O(nlg\ n),\ O(n^2),\ O(n^2 lg\ n),\ O(n^3),\ O(2^n),\ O(n!),\ O(n^n)$

5. Determine the largest size *n* of a problem that can be solved in time *t*, assuming that the algorithm takes *f(n)* milliseconds. *n* must be an integer. (2 points each)

   a. *f(n)* = n, *t* = 1 second **1E³**

   b. *f(n)* = $n \lg n$, *t* = 1 hour **204,094**

I wrote a python script to continually multiply n times lg(n) until it reached above 3.6E^6 (one minute in ms) then subtract 1 to get max integer. Got me close enough to the answer, which I just continually played with until I found the max number to be 204,094 in a calculator.

   c. *f(n)* = n^2, *t* = 1 hour **1897**

d. *f(n)* = n^3, *t* = 1 day **442**

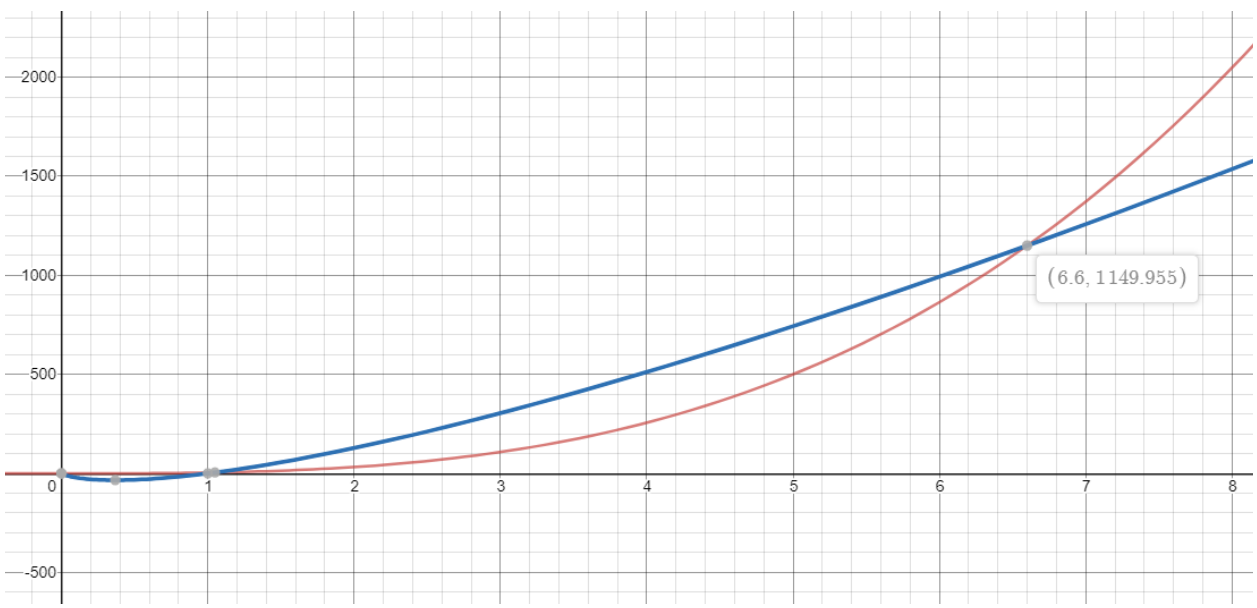
e. *f(n)* = $n!$, *t* = 1 minute **8**

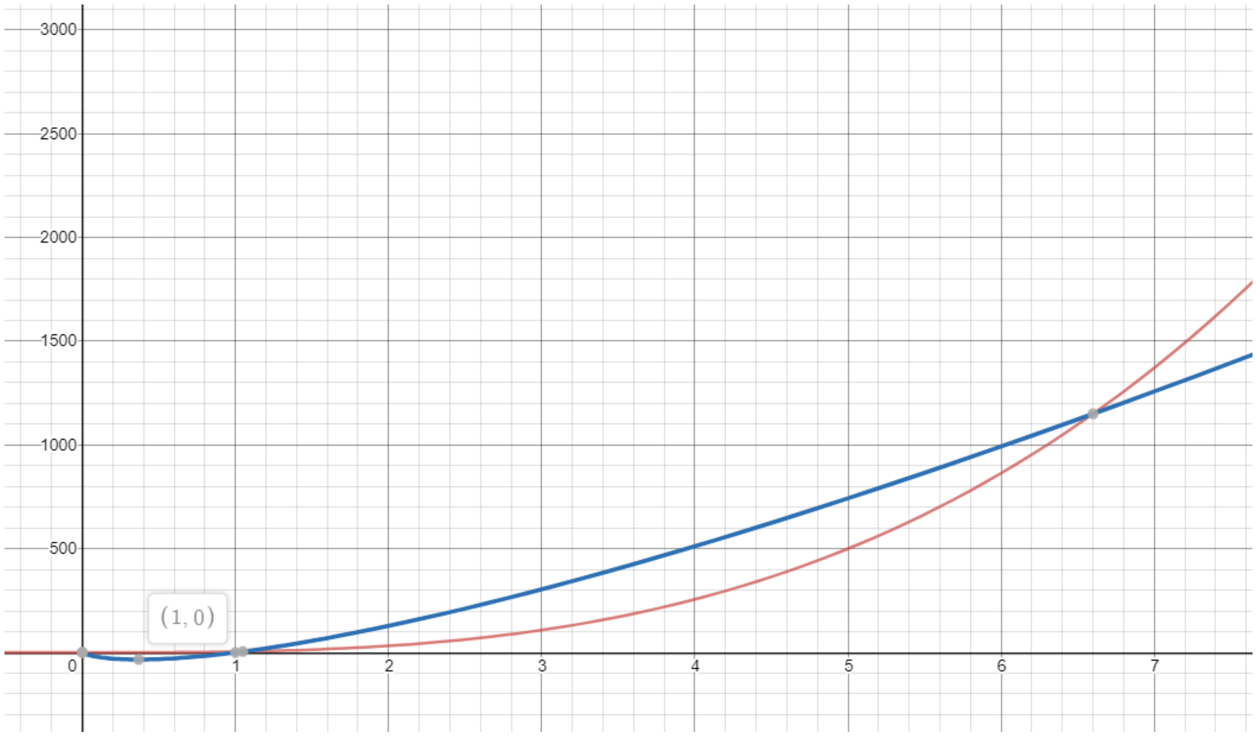
6. Suppose we are comparing two sorting algorithms and that for all inputs of size n the first algorithm runs in $4n^3$ seconds, while the second algorithm runs in $64n\ lg\ n$ seconds. For which integral values of n does the first algorithm beat the second algorithm? $\mathbf{1 \leq n \leq 6}$ $**$ (4 points)

****Technically, it may be** $2 \leq n \leq 6$, since by some calculators the intersection is actually 1.049, making 1 not included in this domain. But, I think the problem simply expects 1, because 1.049 is very close to 1 and Desmos may be off on its estimation.

Explain how you got your answer or paste code that solves the problem (2 point):

By graphing $y = 4x^3$ and $y = 64xlg(x)$, we get two points of intersection:





These two points of intersection denote where the blue function ($y = 64xlg(x)$) takes a *longer time* to process the input, or in other words, where $y = 4x^3$ is faster. These two points of intersection are (1,0) and (6.6, 1150), making the integer domain where the first algorithm ($4n^3$) is faster $1 \leq n \leq 6$.

PLEASE NOTE: Technically, the domain may be $2 \leq n \leq 6$, since by some calculators the intersection is actually 1.049, making 1 not included in this domain. But, I think the problem simply expects 1, because 1.049 is very close to 1 and Desmos may be off on its estimation, and even shows (1, 0) as a point of intersection.

7. Give the complexity of the following methods. Choose the most appropriate notation from among O, Θ, and Ω. (8 points each)

```
int function1(int n) {
    int count = 0;
    for (int i = n / 2; i <= n; i++) {
        for (int j = 1; j <= n; j *= 2) {
            count++;
        }
    }
    return count;
```

```
}
```
Answer: θ(n lg(n))

```
int function2(int n) {
    int count = 0;
    for (int i = 1; i * i * i <= n; i++) {
        count++;
    }
    return count;
}
```
Answer: θ$(\sqrt[3]{\boldsymbol{n}})$

```
int function3(int n) {
    int count = 0;
    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <= n; j++) {
            for (int k = 1; k <= n; k++) {
                count++;
            }
        }
    }
    return count;
}
```
Answer: **θ(**$\mathbf{n^3}$**)**

```
int function4(int n) {
    int count = 0;
    for (int i = 1; i <= n; i++) {
        for (int j = 1; j <= n; j++) {
            count++;
            break;
        }
    }
    return count;
}
```
Answer: **θ(n)**

```
int function5(int n) {
    int count = 0;
    for (int i = 1; i <= n; i++) {
        count++;
    }
    for (int j = 1; j <= n; j++) {
        count++;
    }
    return count;
}
```
Answer: **θ(n)**

I pledge my honor that I have abided by the Stevens Honor System.
Ryan J. Hartman